# DARWIN - INTEGRATED INSTRUMENTATION AND INTELLIGENT DATABASE ELEMENTS

John A. Schreiner*, Phillip J. Trosin†, Cathy A. Pochel‡, and Dennis J. Koga§
NASA Ames Research Center
Moffett Field, California, 94035

**Abstract**

A variety of instrumentation, data acquisition, computer networking, and data presentation systems is being integrated into the wind tunnels of NASA Ames Research Center. The goal of this effort is to increase the variety, quantity, quality, and utility of the information available to the test managers while maintaining production data acquisition rates. In so doing, the efficiency and productivity of wind tunnel testing can be increased. This task is accomplished by networking state-of-the-art data acquisition systems to a central data collection and process management computer over high speed networks. The central process, called ServIO, handles subsystem management and integrates the data from these systems to load a database and file system with test point descriptive information (metadata), and results (data) files. The database/file system is also loaded with predicted and computed results from computer simulation codes. The database and file system are connected to a secure high speed FDDI based network of data analysis and presentation systems. These systems allow the customer to perform queries of the database metadata, retrieve the appropriate results/data files, and analyze and compare these data across test conditions and, eventually, across multiple tests. The first use of this system will be in the NASA Ames 12-Foot Pressure Wind Tunnel to support development wind-tunnel testing in the summer of 1996.

**Nomenclature/Abbreviations**

| | |
|---|---|
| APMS | Ames Power Management System |
| CFD | Computational Fluid Dynamics |
| DARWIN | Developmental Aeronautics Revolutionizing Wind tunnels and Information systems of NASA |
| FCS | Facility Control System |
| FDDI | Fiber Distributed Data Interface |
| IofNEWT | Integration of Numerical and Experimental Wind Tunnels |
| IRT | Infrared Thermography |
| LVS | Laser Vapor Screen |
| MAPPS | Microphone Array Phase Processing System |
| NFS | Network File System |
| PDV | Planar Doppler Velocimetry |
| PSP | Pressure Sensitive Paint |
| RAID | Redundant Array of Inexpensive Disks |
| RAWT | Remote Access Wind Tunnel |
| RPC | Remote Procedure Call |
| SDS | Standard Data System |
| ServIO | Server of Input and Output control information |
| SGI | Silicon Graphics, Inc. |
| SQL | Standard Query Language |
| TDI | Test Dependent Instrumentation |
| TSP | Temperature Sensitive Paint |
| WICS | Wall Interference Correction System |
| WTD | Wind Tunnel Data file |
| WTES | Wind Tunnel Executive System |

## 1.0 Introduction

NASA Ames Research Center has implemented an Integrated Product Team called Developmental Aeronautics Revolutionizing Wind tunnels and Information systems of NASA (DARWIN[1]). DARWIN's products, the prototype technologies for the DARWIN Information System of NASA, provide

* Aerospace Engineer
† Computer Programmer, Sterling Software, Inc.
‡ Mathematician
§ Assistant Chief, Design Cycle Technologies Branch

aeronautical customers with streamlined access to experimental and numerical results before, during, and after wind tunnel tests, in order to increase the efficiency and productivity of wind tunnel testing.

The streamlining of information access is possible due to the rapid advances in information sciences and related technologies. These technologies, including numerical modeling capabilities, computer hardware, instrumentation, information sciences, advanced networking, scientific visualization, and test techniques, have matured to the point where they are changing the definition and characteristic of experimental testing and its role in the design process of new aircraft. The synergistic combination of these technologies offers an approach to increase the confidence in designs and understanding of wind tunnel results, potentially reducing the number of design-and-test cycles in the design process. The potential savings from such reductions are measured in millions of dollars per aircraft design.

The DARWIN Information System of NASA is composed of five major product lines. These are:

1) Remote Access Systems
2) Integrated Instrumentation
3) Intelligent Database
4) Data Visualization
5) Infrastructure Enhancements

This paper will discuss the design and development of two of these products; Integrated Instrumentation and Intelligent Database. In addition, this paper will introduce the concept of the Wind Tunnel Executive System (WTES) which performs such functions as smart scheduling and intelligent system coordination.

## 2.0 Background

NASA Ames Research Center has been a world leader in aeronautical research and wind tunnel testing technology for the past six decades. During this time, the technology of data acquisition, reduction, and analysis has changed dramatically. NASA Ames has continually upgraded its technology and processes to incorporate these changes in an ongoing effort to improve the quality of service and support provided to its customers. The rapid acceleration of the technological revolution since the early 80's has created an information overload producing both an opportunity for breakthrough analysis techniques, and a massive information management workload. In the production wind tunnel environment, the information management workload can prove devastating to productivity.

Consider how the quantity of "on-line" data available for use during a typical production wind tunnel test at NASA Ames has increased over the past decades (Figure 1). On-line data, in this discussion, refers to the *subset* of all the data assembled, acquired, and reduced, which are actually analyzed and studied actively by the researchers. A typical test conducted in 1970 produced approximately 2 Megabytes (MB) of reduced data which were processed on an IBM mainframe computer system. The data were available 24 hours after acquisition, with post test analysis available 6 to 12 months later. By 1980, the quantity of reduced data had increased to 8 MB on a VAX 11-780 computer and the delay reduced to 12 hours. By the late 1980's, global quantifiable image acquisition systems such as Pressure Sensitive Paint (PSP), were beginning to see use. The reduced data quantity grew to 80 MB. Instant access to these data was possible due to local distributed processing. In 1994, the IofNEWT project[2] introduced integrated Computational Fluid Dynamics (CFD) results on-line during testing. The reduced data quantity increased to 3 Gigabytes (GB) with instant data availability. The projection for DARWIN's first production test in the 12-Foot Pressure Wind Tunnel, March 1997, is 10 GB of reduced data on-line with a goal of near-time integrated data analysis and comparison.

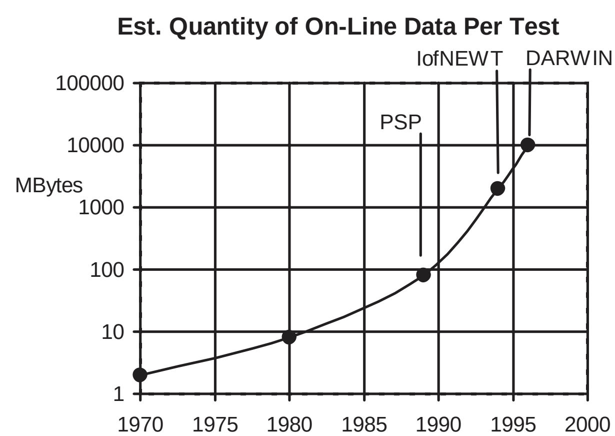


Figure 1.

While data quantities, data acquisition system performance, and analysis processes have improved, the data management and comprehensive data analysis strategies have not kept pace. As a result, most of the data analysis occurs *after* a test is completed. This occurs because there is simply too much data to process using existing analysis strategies, and the variety of information available for analysis requires expertise in too vast an array of computer software, hardware, and interpretive skills to be practical in the fast paced environment of the production wind tunnel. Furthermore, if questions arise during the post-test analysis phase and additional data are required, a new test entry will be needed. This additional entry can produce design delays of up to a year as well as substantial additional cost. In many cases, this additional test entry could be avoided if the data analysis could be completed during the initial test entry.

Under the new aircraft design paradigm which DARWIN offers, a more complete knowledge and understanding of aircraft performance is gained very soon after each data point is taken. As more knowledge is gained, more questions will be raised while there is still an opportunity to acquire the additional data which might hold answers to those questions. More options can be explored within each test, and test plans can be altered if necessary to investigate interesting or unexpected phenomena discovered during analysis.

To achieve these goals, the DARWIN Information System requires a foundation of two primary system functions:

1) Hardware and software integration across multiple platforms
2) Comprehensive information management.

The hardware and software integration systems must work with a diverse array of instrumentation and data acquisition hardware, software, and operating systems. They must be robust, to allow for use in the production wind tunnel environment, yet flexible, to allow interface to research and horizon technology-based data acquisition systems[6-9]. The information management systems must efficiently collect, organize, and publish this information. More than simply publishing the data however, the information management systems must bring together related information to allow for advanced comprehensive data analysis techniques which will allow timely analysis of large quantities of on-line data. The DARWIN program element addressing the hardware and software integration function is called “Integrated Instrumentation”, and the DARWIN program element addressing the information management function is called “Intelligent Database”.

## 3.0 Integrated Instrumentation

The Ames production wind tunnel environment can be generalized into three main system groups as shown in Figure 2. These three system groups must coordinate their actions for successful data acquisition during a wind tunnel test. This coordination activity is performed by the Test Executive.

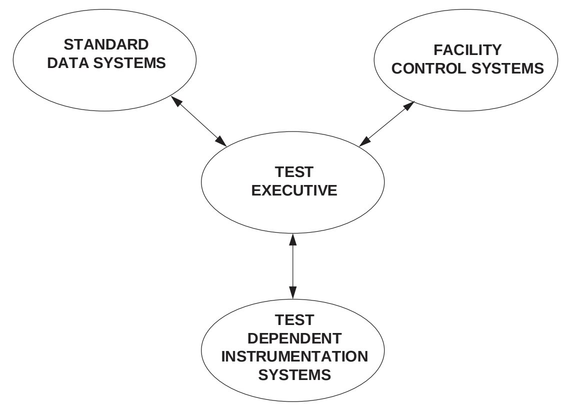


Figure 2.

The Facility Control System[4] (FCS) is responsible for setting the tunnel conditions such as Mach Number, pressure, temperature, as well as model conditions such as model angle-of-attack, side-slip, roll angle, etc. The Standard Data System[5] (SDS) is responsible for measuring and recording wind tunnel conditions, model forces and moments, temperatures, and other conventional production test data measurements. Test Dependent Instrumentation (TDI) systems include Pressure Sensitive Paint[6] (PSP), Planar Doppler Velocimetry[7] (PDV), Laser Vapor Screen (LVS), Infrared Thermography (IRT), Microphone Array Phase Processing System[8] (MAPPS), Liquid Crystal Surface Visualization[9] (LCS), and other new technology systems which are usually employed on selected tests for special measurement requirements. The Test Executive refers generically to the person or system which coordinates the operation of the SDS, FCS, and TDI systems. At NASA Ames, the executive function is currently performed by the Test Engineer.

Of primary concern to the integrated instrumentation task is the coordination and management of the TDI and SDS systems. To address these concerns, the integrated instrumentation systems must perform three main functions:

1) Coordinate communications and actions among the diverse array of instrumentation and data acquisition hardware, software, and operating systems comprising the TDI and SDS.
2) Manage the inter-reliance and timing that many of the new TDI systems require.
3) Provide an open architecture growth path for future expansion to include new TDI systems.

This third function is based upon the desire to include not just the traditional pool of “standard” data sensing systems which have been used for decades in production wind tunnels, but also the “break-through” technologies which use optical and acoustic imaging of the test model and surrounding environment (e.g. PSP, PDV, MAPPS, LCS, etc.). These technologies, while offering terrific increases in data collection per unit time, have not matured to easily fit the production wind tunnel environment of sustained high data point rates over many days of continuous and automated operation. In addition, these systems have not evolved standard analysis tools and techniques for reducing their results. This causes a lack of standardization in result data format (e.g. GIF, TIFF, FITS, CGNS, etc.), size, and type (e.g. binary, ASCII, etc.).

DARWIN has implemented a software process called ServIO to perform the integration task. A detailed discussion of the ServIO process is presented in the following section.

## ServIO

At the most fundamental level, ServIO represents an information access and transmittal layer between the SDS, TDI, and the DARWIN Database and File System (Figure 3).

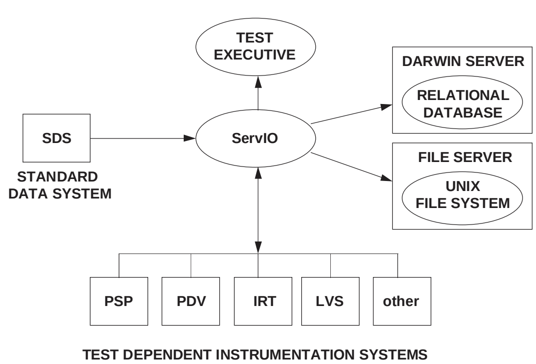


Figure 3.

Its basic charter includes providing SDS data in near-time to support various TDI systems, monitoring and displaying the status of the data acquisition process of each system, and making final results available from the SDS and TDI systems as quickly as possible, by adding references to them (metadata) in the DARWIN Database, and placing SDS results files in the file system.

ServIO is a fully automated process whose operation and actions are governed by a user generated control file. This control file contains all the information needed to establish communications with the SDS and key information about each of the TDI systems. The control file identifies which TDI’s are to be supported and what additional data items each TDI system requires from the SDS in order to complete that system's data reduction. The control file also contains information describing the results files which will be produced by each of the TDI systems. This information is needed by ServIO to construct proper DARWIN Database Records (described later in this paper).

ServIO operations can be generalized into three main categories:

1) SDS communications and system monitoring.
2) Graphical system status display.
3) Information collection, database record building, and file system updating.

A more detailed discussion of these three functions follows.

**SDS Communications and System Monitoring:** The SDS supports access to near-time results via a software process called LinK_Serve. The LinK_Serve process can become overwhelmed by data requests if the number of on-line clients becomes too large. In addition, many of the TDI systems require the same information from the SDS causing unnecessary redundancy and workload on the server. To avoid this, ServIO acts as a single point of contact with the SDS for near-time retrieval of data items that are shared by all the TDI systems and the DARWIN Database. This communication with the SDS is necessary because many of the TDI systems require test point index values (e.g. run and sequence numbers), tunnel conditions, and other measured results from the SDS in order to complete their data acquisition and reduction processes. For example, a PSP image analysis system will require run and sequence numbers, information describing the wind tunnel model configuration and angle-of-attack (to identify related wind-on and wind-off images), and SDS transducer measured pressure coefficients (Cp's) for *in-situ* calibration of the PSP data. Without the SDS data, PSP would be unable to reduce its data and the customer would not have immediate access to PSP results.

Near-time access to the SDS results is configured in the ServIO control file. After reading and validating the control file, ServIO launches a process called Standard Data System Link (SDSLK) which establishes a connection with LinK_Serve. LinK_Serve employs sockets to establish bi-directional communications (client-to-server to establish communications and to issue requests for data transmittal, and server-to-client to send requested results back to the ServIO host). Data is transmitted in the form of "packets". As data packets arrive from the SDS, SDSLK interprets the packet contents and writes the data as ASCII files. As these ASCII files are received, ServIO produces, saves, and transmits Wind Tunnel Data (WTD) files, as shown in Figure 4. The WTD files are transmitted to each TDI system, and contain the subset of results necessary for each TDI system to perform its data reduction and analysis.

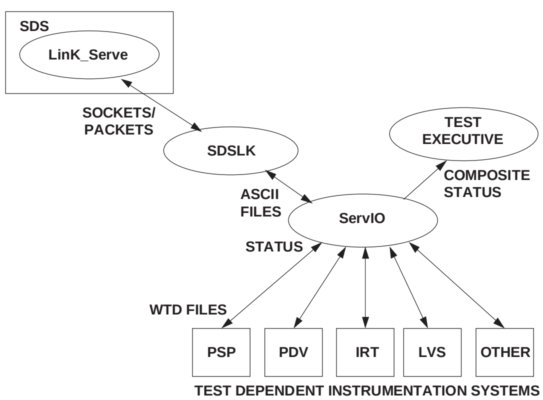


Figure 4.

ServIO monitors the status of the data acquisition process of each TDI system, and provides a convenient graphical display of that status. ASCII status files are written by each TDI system as the status of its data acquisition process changes. These ASCII files are written by each TDI system to a Network File System (NFS) mounted disk directory published from the ServIO host. The use of NFS allows the TDI developers to implement software which accomplishes ServIO communications simply by writing files to disk. The file saves appear local to the client's application but are actually remotely mounted by the operating system. This allows the TDI developers to use any commercial software running on any computer system (Apple, PC Clone, HP, Sun, SGI, etc.) and operating system (UNIX™, VMS™, OS9™, OS/2™, DOS™, Windows™ 3.x, Windows NT™, System™ 7.x, etc.), without modification, as long as the system can perform an NFS mount operation.

ServIO reads the ASCII files and extracts the TDI system state for use in its composite logic and display. Status values were chosen to support a simple five-state model of the data acquisition process:

1) The system is **turned off**
2) The system has encountered an **error**
3) The system is busy **acquiring data**
4) The system has finished acquiring data (but need not have finished recording it). It is **okay to move** to the next set of test conditions.
5) The system is **ready to acquire the next data point**

ServIO combines the status information it receives from the various TDI systems to form a composite status value which is sent to the Test Executive. Composite status values typically represent the "least ready" status signal received from the active TDI systems at any given time. Composite status values indicate to the Test Executive such things as whether to hold at the present test conditions, whether it is okay to move to the next test condition, and whether it is okay to begin to acquire the next data point. The Test Executive would use ServIO's composite status indicator in conjunction with similar signals from FCS and SDS to determine the next action for tunnel operations.

**ServIO Display**: An example of the ServIO display is shown in Figure 5. For each TDI system identified in the control file, ServIO displays a status button which presents the most recent status information received from that system. If a TDI system is turned off, the button for that system is colored gray. If a TDI system reports an error condition, its status button displays that there has been an error and flashes (color alternates between red and yellow). Other status information received from a TDI system are indicated in the display with color and button size change. This facilitates monitoring of the display screen from across the wind tunnel control room.

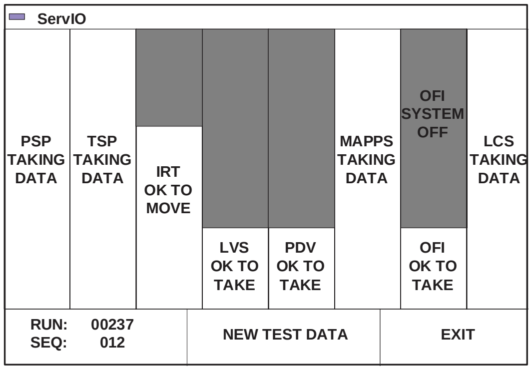


Figure 5.

A basic level of control is provided via mouse-based interactions with the various graphical buttons presented in the ServIO display. A manual override is provided for clearing a hung or incorrect status value reported by an individual TDI system. Picking a TDI system's status display button with the mouse cursor causes the status to cycle among "off", "busy taking data", "okay to move model", and "okay to take data" states. The color of the picked button cycles along with the displayed text from gray, to red, to yellow, to green, respectively. These changes are local to ServIO. The TDI systems are not affected, nor are they informed of the action that has taken place within ServIO. The override provides a means of forcing new composite status information to be sent to the Test Executive. In this way, the Test Executive can be instructed to move on to the next data point rather than be stalled by a "hung", malfunctioning, or disconnected TDI system.

Lastly, the display shows ServIO's data acquisition mode. ServIO provides "New Test Data" and "Recomputed Test Data" acquisition modes. In New Test Data mode, ServIO manages all data point-by-point as it is acquired. In Recomputed Test Data mode, prior test data is transferred to ServIO in batch. Neither the SDS nor TDI systems are acquiring new data in Recomputed Test Data mode.

**Information Collection and Database Record Building:** ServIO is responsible for assembling the appropriate data from the SDS with knowledge of the results generated by the TDI systems to produce entries in the DARWIN Database. The entries are passed from ServIO to the database by means of an ASCII file called a Namelist (NML). Each NML created by ServIO contains a complete metadata record for one data point. The NML files are saved by ServIO in a directory accessible by a script launched from the DARWIN Server (database host platform). From the DARWIN Server, the database is able to load the NML's on a regular basis. Once the NML's are loaded into the database, the data are immediately available to all on-line systems.

In addition to adding records to the database, ServIO can use the NML mechanism to change, and delete database records. A key feature of this loading mechanism is that the ServIO process is able to create NML's at a pace matching the data acquisition process in the production wind tunnel while the database loading process is performed at a rate chosen to optimize database server performance (i.e., loading of NML's will occur during low database query activity). The NML mechanism therefore, acts as a buffer to allow the data acquisition and database update processes to run asynchronously. In this way, the timing of each process can be optimized without affecting the performance of the other. Further details of the Namelists are covered in the database section of this paper.

**Software Implementation:** ServIO comprises roughly 4,000 lines of FORTRAN code (including in-line documentation). It runs on Silicon Graphics, Inc. (SGI) IRIS$^{TM}$ workstations under IRIX$^{TM}$ 5.3 and uses SGI's gl graphics library to create its displays. Network File System (NFS) is used by the TDI systems and by the ServIO host to allow sharing of status, WTD files, and Namelists across Ethernet networks. NFS was chosen because of its availability and low cost for a wide variety of platforms. In addition, NFS allows a straightforward implementation of the system status and WTD file sharing concepts, with minimal impact on software development teams for the TDI systems.

## 4.0 Intelligent Database

The DARWIN Intelligent Database is responsible for organizing and providing searchable access to the top level data from the SDS, TDI, CFD, and related test data systems. The database is designed to contain relational fields of metadata. These metadata consist of both data and pointers to data. The pointers can be host computer addresses (IP number, URL, Internet address), disk directory paths, or file names. The database is designed to meet several requirements including:

1) Allow metadata searches using Standard Query Language (SQL).

2) Allow frequent metadata record updates from ServIO (as proxy to the TDI and SDS systems) via the NML mechanism.
3) Allow simple metadata record updates by test managers.
4) Allow interface to Web Server queries as defined in the Remote Access and Intelligent Visualization products[3]
5) Run on the DARWIN Server described by Korsmeyer[3].

The Design and Operation of the DARWIN Database model are described in the following sections.

**The Database Design:** The database design is a hybrid of a flat-file and a minimally normalized Relational Database Model (RDBM). The current implementation has four relational tables which correspond to the four primary indexing values used in the Ames Wind Tunnels. These are:

1) The Test table composed of one record per test.
2) The Configuration Table composed of one or more Configuration Records per Test Record.
3) The Point Table composed of many Point Records per Configuration Record.
4) The File Table composed of one or more File Records per Point, Configuration, or Test Record.

The File Records contain the file name pointers to the results files held in a UNIX file system as shown in Figure 6. The host address of the file system and directory paths of the results files are contained in the Test Table.

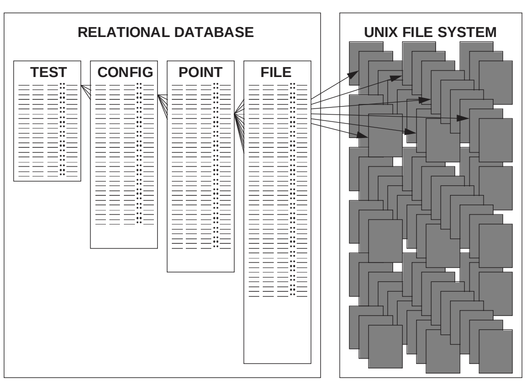


Figure 6.

The database tables contain descriptive information (metadata) which the user will query to identify the results of interest. In this model, the user can query the Test Table for the specific wind tunnel test of interest, followed by queries of the Configuration, Point and File Tables to locate the specific data from the selected tests. Much of the metadata in the Point Table is essential design cycle information such as Mach Number, Angle-of-Attack, Dynamic Pressure, Lift Coefficient, Drag Coefficient, etc. This will allow the user the locate data via an SQL query such as;

*"select RunNo, PointNo, CL from Point where CL>1.2"*

This query will return the Run and Sequence Numbers and Lift Coefficient for all data points where the lift coefficient is greater than 1.2. This query will be performed via a graphic user interface (GUI) developed under the task described in the paper by Korsmeyer[3]. The purpose of the GUI is to isolate the user from SQL syntax and to provide an intuitive interface for database searches. Global results such as PSP, PDV, and CFD would then be located with a query of the File Table. For example, suppose the above query returned a listing which identified RunNo 234 and PointNo 11 as having a value of CL = 1.6. If the user determines that such a CL deserves further investigation, the user can choose to review other available results. For example, the user can choose to view the surface pressures for that point as acquired by the PSP system. The appropriate SQL statement would look as follows;

*" select FileName from File where RunNo=234 and PointNo=11 and ID="PSP" "*

This query will return the PSP system results file names for Run 234 and sequence 11. Again, the GUI will isolate the user from the SQL syntax requirements. In addition, the GUI will perform the file transfer, launch the appropriate viewer, and present the PSP data on the user's client computer. The creation of the SQL, retrieval of the paths and filenames, and file transfer protocols, would be transparent to the user via the GUI.

The database metadata records are loaded by a script resident on the DARWIN Server and launched every n time cycles as a UNIX "cron" process. The script checks for the existence of NML's on the ServIO process host system. If the script process detects files with the extension .nml (which is the extension attached to all NML's), it moves the files to the DARWIN Server (deleting the originals on the ServIO host) and places them in a temporary directory. The script then launches a database loading script which performs the database update.

The database loading script is responsible for updating the database, as well as performing error detection and notification. The script is able to distinguish the destination database and table from the contents of the NML. The script maintains a transaction log which contains a history of database updates and errors. After the database loading script has successfully completed the update, it deletes the NML's from the temporary directory. This cycle is repeated with each execution of the cron process scripts.

The NML is a field delimited ASCII string. The individual fields of this string correspond to the metadata fields of the database records. There are four types of NML corresponding to each of the four tables in the database. While the Namelist is typically created by ServIO as described in the previous section, it can also be created by other TDI systems, or manually produced by user interaction.

The rules for generating an NML are designed to allow the user to manually produce the NML string. In this way, a test manager could produce a new entry in the database by placing a manually created NML in the ServIO process host directory. The scripts launched from the DARWIN Server will move this NML along with those created by the ServIO process. Once the script has moved the NML, the database update will take place. Thus, the NML provides a simple mechanism for placing metadata into the DARWIN Database.

In addition to the NML mechanism, there are other more flexible methods for maintaining entries in the database. The SQL interface is available, as well as GUI tools which run as layers on top of SQL. These tools allow an experienced UNIX user and database administrator unlimited flexibility in database management.

The DARWIN Database contains values from SDS but does not contain the final results generated by the TDI systems. By using pointers to the TDI results, database loading can occur at SDS production data rates while the TDI systems move their results asynchronously to the file system as soon as they are complete. As a result, the database query will return data from the most recent data points acquired while the detail information from the TDI systems can follow as soon as the TDI systems place them on the file system. This approach simplifies inclusion of "late" results arriving from TDI systems with computationally intensive and time consuming data reduction processes.

**Implementation:** The DARWIN Database is built using the Sybase SQL Server™ relational database, Version 10.0.2.4. The server is installed on an SGI Challenge™ S Server running IRIX™ 5.3 operating system. The SGI system has a single 150 MHz R4400 Processor with 64 MB RAM, 12 GB of differential SCSI disk space, and a FDDI network interface.

**The File System:** The file system provides a flexible directory structure in which the TDI systems place results. The file system location is identified in the DARWIN Database by host address (URL, IP address, or fully qualified Internet address) allowing the entire file system to be moved to another machine. The DARWIN Database host address field can be easily updated to point to the new host machine. This method is extendible to allow multiple File system hosts to be referenced in the DARWIN Database allowing for virtually unlimited data storage and locating capability without affecting the remainder of the database file locator scheme.

**Implementation:** The File system coresides with ServIO on an SGI Challenge™ S Server running IRIX™ 5.3 operating system. The SGI system has a single 150 MHz R4400 Processor with 64 MB RAM, 10 GB of differential SCSI disk space, and a FDDI network interface. There will be one file system located in each of the Ames Wind Tunnels supported by the DARWIN Network described by Koga[1] and Korsmeyer[3].

## 5.0 System Operation

Figure 7 shows the complete data path from acquisition to database update and file system loading. The steps in the process are numbered in the figure and are described below:

1. The SDS and TDI systems take data.
   The SDS sends LK data packet to ServIO.
2. ServIO sends WTD files to the TDI systems which complete data acquisition and begin data reduction.
3. The TDI systems update ServIO status.
4. ServIO creates Namelist (NML) files.
   ServIO updates the executive to "take next point status".
   Script moves NML's from ServIO host to the

DARWIN Database host.
Script updates DARWIN Database.

--- synchronous return to step 1---
--- asynchronous proceed ---

5. Results files are written to the file system when available.

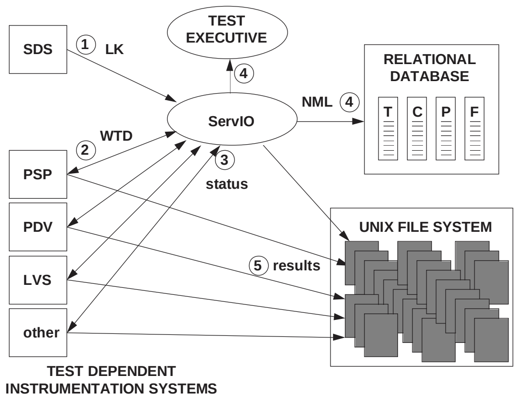


Figure 7.

### 6.0 Wind Tunnel Executive System

The Wind Tunnel Executive System (WTES) is a new proposal being considered by the NASA Ames Research Center's Aeronautical Test and Simulation Division, Code AO, for implementation with the DARWIN information systems. The WTES will provide communication among the various data systems (SDS, FCS, TDI), and provide valuable test guidance information to the Test Engineer.

**Phase I:** In its first implementation (phase I), the primary objective will be to automate the communications and management tasks currently performed by a human (the Test Engineer). In this model, the WTES will perform central automated coordination of data acquisition and system status among the SDS, FCS, and ServIO systems so that the data can be acquired and indexed with proper timing and synchronization. This includes providing the low-level signals required by the TDI systems (such as a switch closure) to signal start of data acquisition, emergency stop, etc.

**Phase II:** As the WTES expands in follow on implementations, it will take on a more central role, as shown in Figure 8. The WTES will communicate with the FCS, The Wall Interference Correction System (WICS), the Ames Power Monitoring System (APMS), the SDS, and ServIO (as proxy for the TDI systems). The APMS is an extermely useful information source for the WTES. The APMS evaluates the proposed run schedule against other power demands at the center, facility constraints, and by a power pricing algorithm. In this central role, therefore, the WTES will have access and control over a more comprehensive suite of systems allowing it more authority and capability to effect wind tunnel test efficiency.

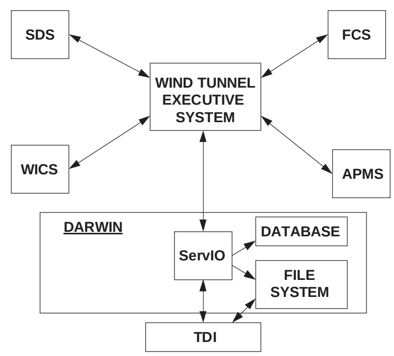


Figure 8

**Phase III:** In the long term plan, expert systems will be incorporated into the WTES to provide the test director and remote customer (via DARWIN) an analysis of current test status and recommended options to guide the test for maximum efficiency and data rates. The WTES will manage data acquisition requests between SDS, FCS, TDI, and the test director. In addition, it will integrate the test schedule and objectives, to provide a knowledge base to serve the expert systems. The fully developed WTES would provide the test engineer with the flexibility to make run schedule modifications, prioritize options to optimize the test schedule, and have instant visual access to health and status information of all systems.

### 7.0 Results and Discussion

DARWIN is currently deploying an alpha version of the software and hardware described in this paper. The individual systems and software have been tested under simulated conditions and have performed their functions as designed. During the month of July 1996, this alpha system will be used during a four week production wind tunnel test in the Ames 12-Foot Pressure Wind Tunnel (PWT). At that time, the systems will be exposed to 24-

hour per day operations with on-line data acquisition and database query.

The experience gained during this alpha test will be used to define Beta testing. Beta testing of the system is scheduled to occur during the month of September 1996, on a second production wind tunnel test in the Ames 12-Foot PWT.

The system will be fully operational for customer support of production wind tunnel testing during the month of March 1997. Once again, this test will take place in the 12-Foot PWT in support of the Advanced Subsonics Technology (AST) program.

As the alpha and beta testing progresses, hardware and software changes will be made as needed. In addition, a list of potential changes is being compiled. Some of the current proposed changes are listed below:

<u>ServIO:</u>

1) Changing ServIO's graphics base from gl to Motif to enhance the system's portability.
2) File sharing via NFS may require too much system overhead to be practical in high system activity testing. A switch to Remote Procedure Calls (RPC's) or sockets will be considered once the developers of the TDI systems complete their implementations.
3) A more sophisticated communication method could allow ServIO real-time "heart beat" monitoring of TDI systems to determine if a TDI system has "crashed." In the near term, adding a time-out concept for the TDI systems is being considered. Establishing proper criteria for such time-outs is complicated by the dynamic state of TDI system development.

<u>Database:</u>

1) Upgrade of the DARWIN Server System and file system to include error correcting RAID for increased data integrity.
2) Upgrade of the DARWIN Server to a multiprocessor system to increase the database search speed for multiple clients.

## 8.0 Concluding Remarks

Exciting innovations are taking place in sensor technology, and powerful new wind tunnel measurement techniques are emerging more rapidly than ever before. Through the development of the DARWIN Information System, these advances can be integrated and exploited during production wind tunnel testing in order to reduce new aircraft design cycle time, ultimately reducing new design "time to market".

This paper has described two of the five elements of the DARWIN Information System which will interface the emerging sensors and new test techniques. By the spring of 1997, the system will become a permanent part of production wind tunnel testing conducted in the NASA Ames 12-Foot Pressure Wind Tunnel.

Related information about the DARWIN Intelligent Database, Integrated Instrumentation, the DARWIN project, and the DARWIN Integrated Product Team can be found on the world wide web at *http://www.darwin.arc.nasa.gov/* .

## Acknowledgments

The authors would like to thank the Database Development team of Recom Technologies; Sterling Software, Inc. employees Dexter Hermstad, and Gary Sorlien; DARWIN team member Dr. David Korsmeyer; and Design Cycle Technologies Branch Chief Mike George for their assistance, contributions, and support.

## References

1. Koga, D. J., Korsmeyer, D. J., Schreiner, J. A., DARWIN Information System of NASA - An Introduction, AIAA Paper No. 96-2249, AIAA 19th Advanced Measurement and Ground Testing Technology Conference, New Orleans, LA, June 1996.

2. Koga, D. J., Schreiner, J. A., Buning, P. G., Gilbaugh, B. L., George, M. W., Integration of Numerical and Experimental Wind Tunnels (IofNEWT) and Remote Access Wind Tunnel (RAWT) Programs of NASA, AIAA Paper No. 96-2248, AIAA 19th Advanced Measurement and Ground Testing Technology Conference, New Orleans, LA, June 1996.

3. Korsmeyer, D. J, Walton, J. D., Gilbaugh, B. L., DARWIN - Remote Access and Intelligent Data Visualization Elements, AIAA Paper No. 96-2250, AIAA 19th Advanced Measurement and Ground Testing Technology Conference, New Orleans, LA, June 1996.

4. Staff, NASA Ames Research Center, 12-Foot Pressure Wind Tunnel Training Manual, NASA XM No. 306-9291-XM1, NASA Ames Research Center, 1996.

5. Rogers, G., Antosiak, R., Information System Product Specification for Aerodynamics Division Standardized Data System, TN-91-8012-000-3, Sterling Federal Systems, Inc., Oct. 31, 1991.

6. McLachlan, B. G., Bell, J. H., Pressure Sensitive Paint in Aerodynamic Testing, Experimental Thermal and Fluid Science, Elsevier Science, Inc., 1995; 10:470-485.

7. Watts, M. E., Mosher, M., Barnes, M., Microphone Array Phased Processing System (MAPPS), AIAA Paper No. 96-1714, 2nd AIAA/CAES Aeroacoustic Conference, May 1996.

8. McKenzie, R. L., Measurement Capabilities of Planar Doppler Velocimetry Using Pulsed Lasers, Applied Optics, Feb 20, 1996 issue.

9. Reda, D. C., Wilder, M. C., Crowder, J. P., Simultaneous Full Surface Visualization of Transition and Separation using Liquid Crystal Coating, AIAA Paper No. 96-2182, 19th AIAA Advanced Measurement and Ground testing Technology Conference, New Orleans, LA, June 1996.